ANALYTIX
AUDIT

26/08/2022

# PREMIUM AUDIT

| | |
|---|---|
| NAME OF PROJECT | MIKAWA INU |
| TOKEN ETH ADDRESS | 0x36538cef06ff56bda2e8d41ec62137458a7f4a91 |
| WEBSITE / TELEGRAM | N/A |

DISCLAIMER: PLEASE READ FULL AUDIT

# IMPORTANT DISCLAIMER

I. **NOT RESPONSIBLE**

Analytix Audit holds no responsibility for any actions from the project in this audit.

II. **NOT GUARANTEE**

Analytix Audit in no way guarantees that a project will not remove liquidity, sell off team tokens, or exit scam.

III. **INFORMATION**

Analytix Audit researches and provides public information about the project in an easy-to-understand format for the common person.

IV. **AUDIT AGREEMENT**

This audit agreement does not guarantee ANY illicit actions by the project team and does not serve as an advocation for the project.

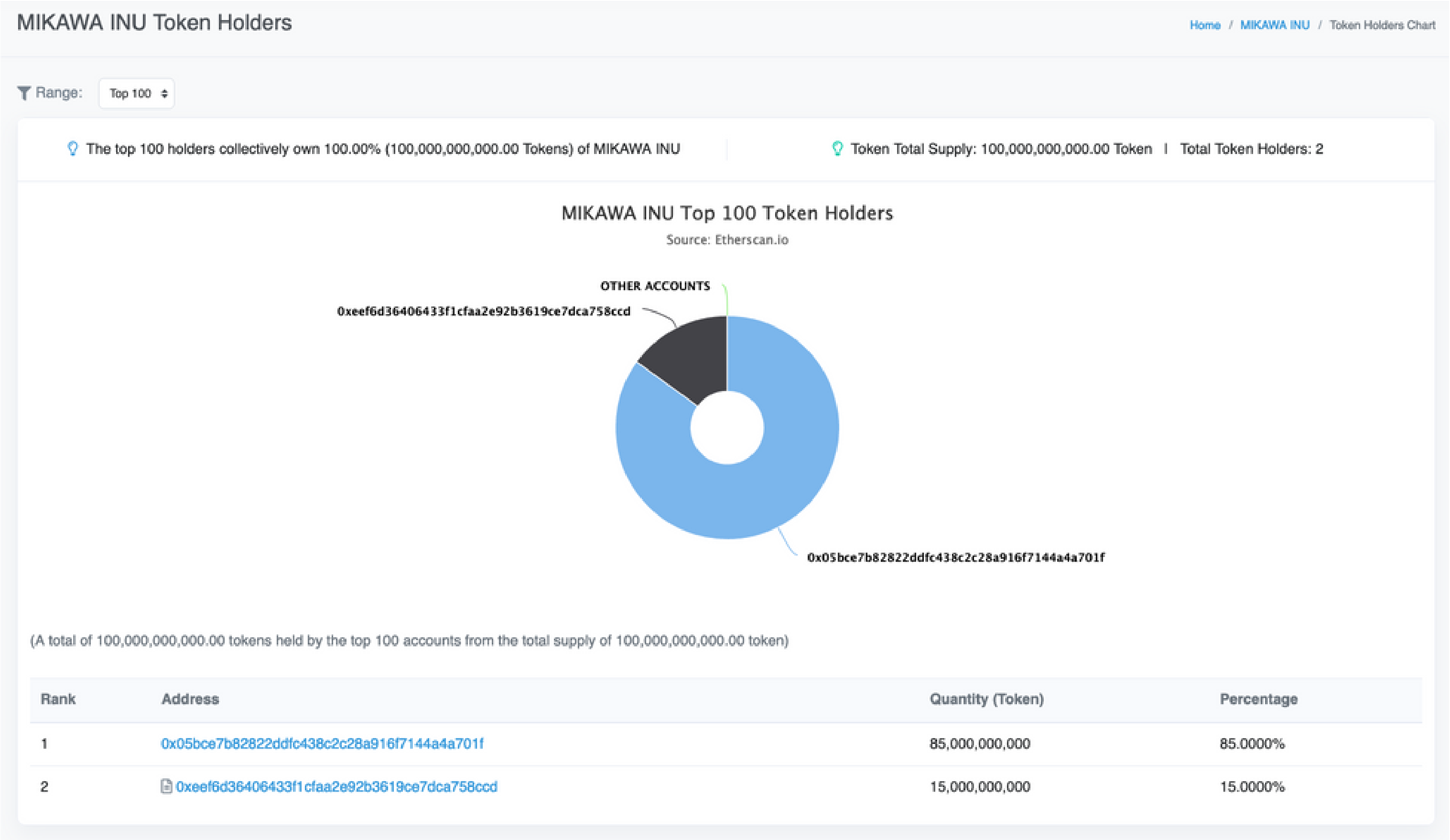
V. **NO ADVICE**

Analytix Audit in no way takes responsibility for any losses, not does Analytix Audit encourage any speculative investments.

VI. **DISCLAIMER**

The information provided in this audit is for information purposes only and should not be considered investment advice. Analytix Audit does not endorse, recommend, support, or suggest any projects that have been audited.

# TOKENOMICS - OVERVIEW



MIKAWA INU Token Holders

Home / MIKAWA INU / Token Holders Chart

Range: Top 100

The top 100 holders collectively own 100.00% (100,000,000,000.00 Tokens) of MIKAWA INU

Token Total Supply: 100,000,000,000.00 Token | Total Token Holders: 2

MIKAWA INU Top 100 Token Holders

Source: Etherscan.io

OTHER ACCOUNTS
0xeef6d36406433f1cfaa2e92b3619ce7dca758ccd
0x05bce7b82822ddfc438c2c28a916f7144a4a701f

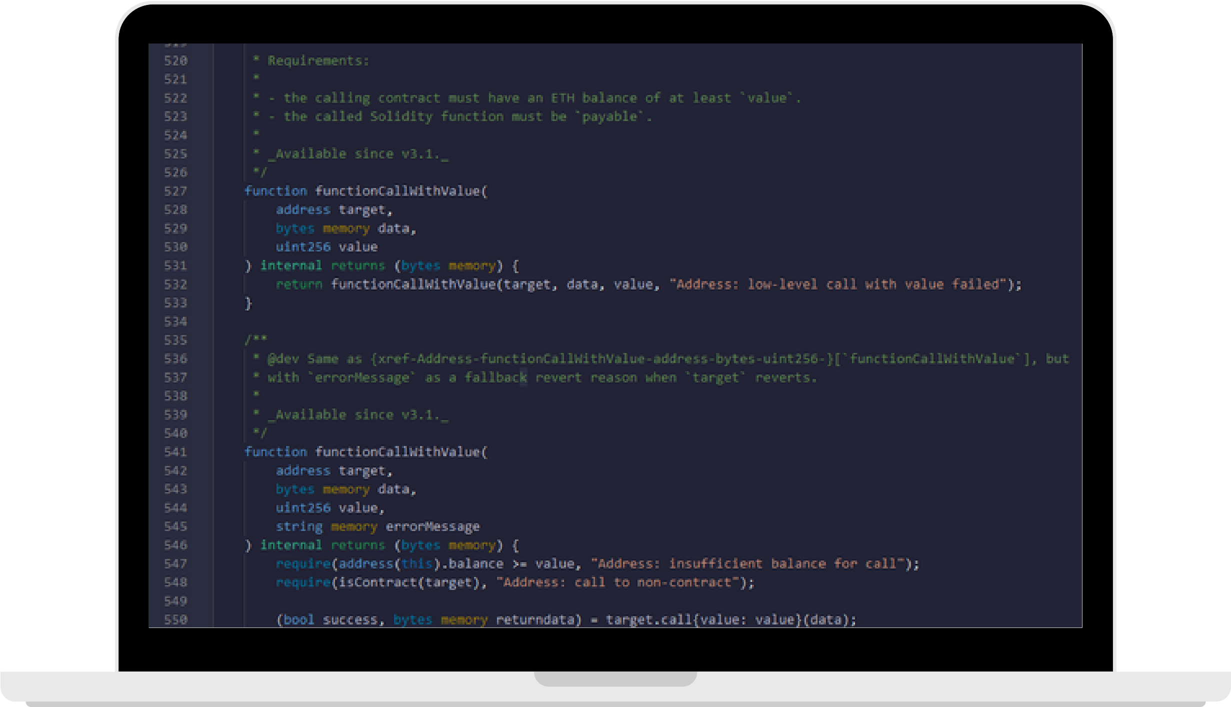
(A total of 100,000,000,000.00 tokens held by the top 100 accounts from the total supply of 100,000,000,000.00 token)

| Rank | Address | Quantity (Token) | Percentage |
|---|---|---|---|
| 1 | 0x05bce7b82822ddfc438c2c28a916f7144a4a701f | 85,000,000,000 | 85.0000% |
| 2 | 0xeef6d36406433f1cfaa2e92b3619ce7dca758ccd | 15,000,000,000 | 15.0000% |

More than 15% of tokens is unlocked

(This comment refers to when the audit was carried out maybe team locked tokens later)

# CONTRACT CODE - PRIVILEGES



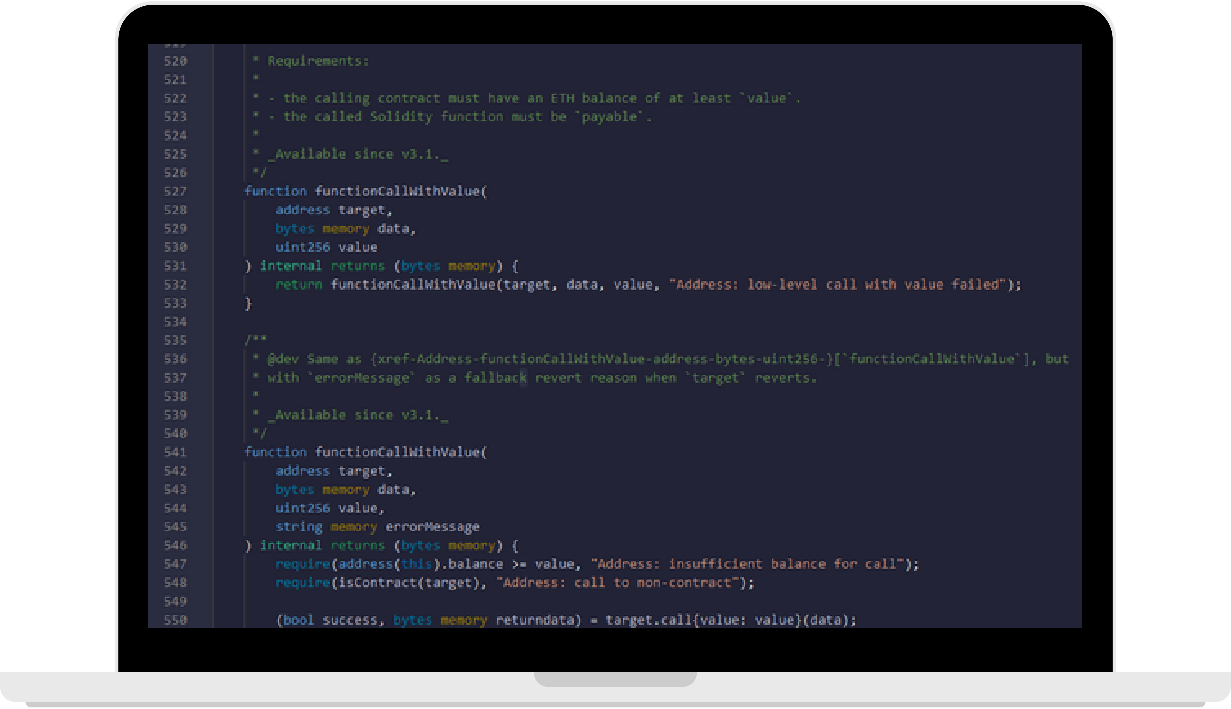
```
     * Requirements:
     *
     * - the calling contract must have an ETH balance of at least `value`.
     * - the called Solidity function must be `payable`.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value
    ) internal returns (bytes memory) {
        return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
    }

    /**
     * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
     * with `errorMessage` as a fallback revert reason when `target` reverts.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value,
        string memory errorMessage
    ) internal returns (bytes memory) {
        require(address(this).balance >= value, "Address: insufficient balance for call");
        require(isContract(target), "Address: call to non-contract");

        (bool success, bytes memory returndata) = target.call{value: value}(data);
```

- **Mint:** Owner cannot mint new tokens
- **Fees:** Owner can't change fees over 25%
- **Trading:** **Owner can change trading status**
- **Max Tx:** Owner can change maxTx with reasonable limits
- **Max Wallet:** Owner can change max wallet with reasonable limits
- **Blacklist:** Owner cannot blacklist wallet

# Owner Privileges:



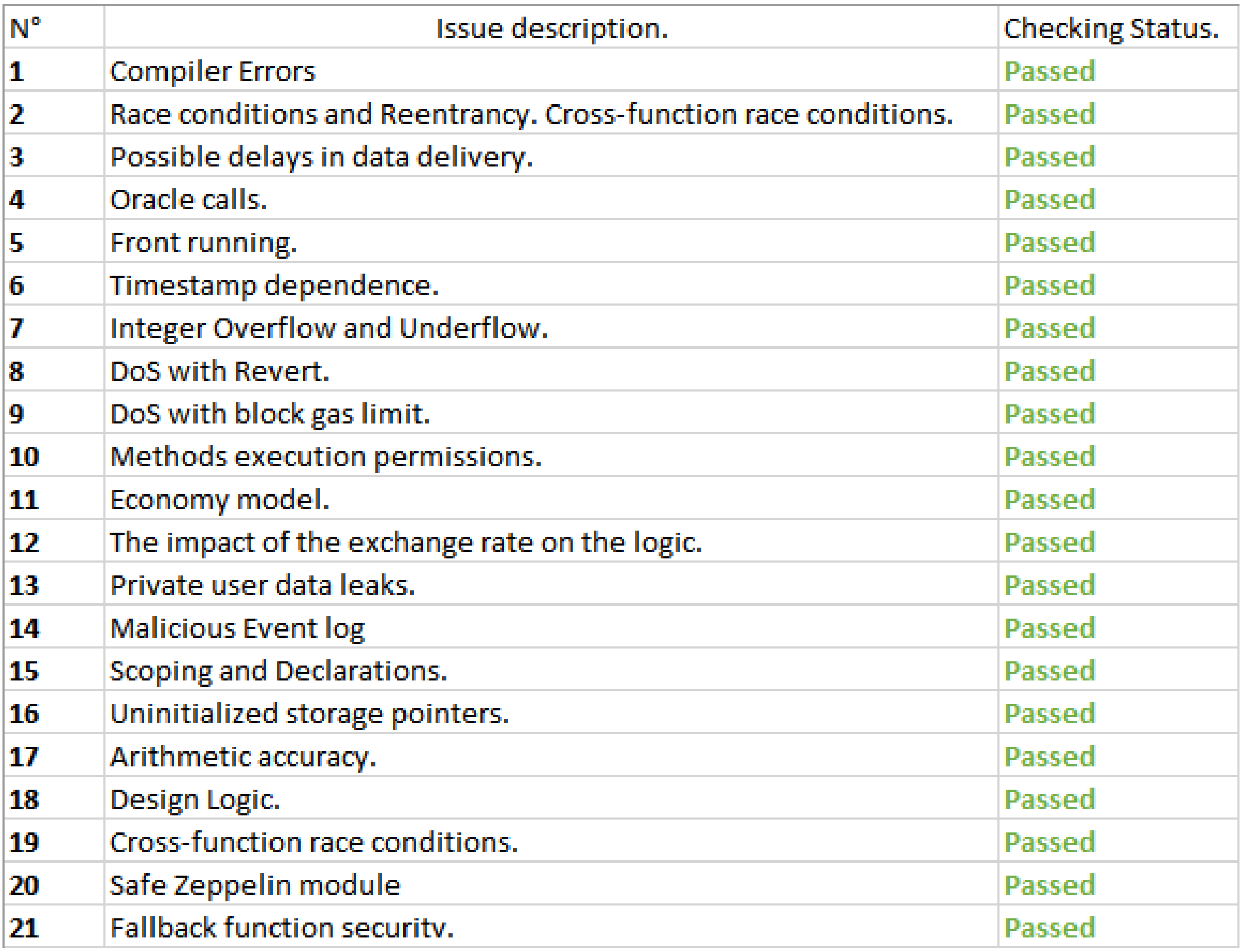
```
     * Requirements:
     *
     * - the calling contract must have an ETH balance of at least `value`.
     * - the called Solidity function must be `payable`.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value
    ) internal returns (bytes memory) {
        return functionCallWithValue(target, data, value, "Address: low-level call with value failed");
    }

    /**
     * @dev Same as {xref-Address-functionCallWithValue-address-bytes-uint256-}[`functionCallWithValue`], but
     * with `errorMessage` as a fallback revert reason when `target` reverts.
     *
     * _Available since v3.1._
     */
    function functionCallWithValue(
        address target,
        bytes memory data,
        uint256 value,
        string memory errorMessage
    ) internal returns (bytes memory) {
        require(address(this).balance >= value, "Address: insufficient balance for call");
        require(isContract(target), "Address: call to non-contract");

        (bool success, bytes memory returndata) = target.call{value: value}(data);
```

function renounceOwnership() public virtual onlyOwner
function transferOwnership(address newOwner) public virtual onlyOwner
function lock(uint256 time) public virtual onlyOwner {
function openTrade() external onlyOwner {
function stopTrade() external onlyOwner {
function includeToWhiteList(address[] memory _users) external onlyOwner {
function excludeFromReward(address account) public onlyOwner()
function includeInReward(address account) external onlyOwner() {
function excludeFromFee(address account) external onlyOwner {
function includeInFee(address account) external onlyOwner {
function excludeFromTxLimit(address account, bool _value) external onlyOwner
function prepareForPresale() public onlyOwner
function afterPresale() public onlyOwner
function setAllBuyFeePercentages(uint256 liquidityFee, uint256 marketingFee)
external onlyOwner()
function setAllSaleFeePercentages(uint256 liquidityFee, uint256 marketingFee)
function setMaxTxAmount(uint256 _mount) external onlyOwner()
function setNumTokensSellToAddToLiquidity(uint256 __minimumTokensBeforeSwap) external onlyOwner()
function setMarketingAddress(address _marketingAddress) external onlyOwner()
function setSwapAndLiquifyEnabled(bool _enabled) public onlyOwner
function setExcludedFromWhale(address account, bool _enabled) public onlyOwner
function  setWalletMaxHoldingLimit(uint256 _amount) public onlyOwner
function setSwapAndLiquifyEnabled(bool _enabled) public onlyOwner
function setExcludedFromWhale(address account, bool _enabled) public onlyOwner     function
setWalletMaxHoldingLimit(uint256 _amount) public onlyOwner

# Issue Checking:

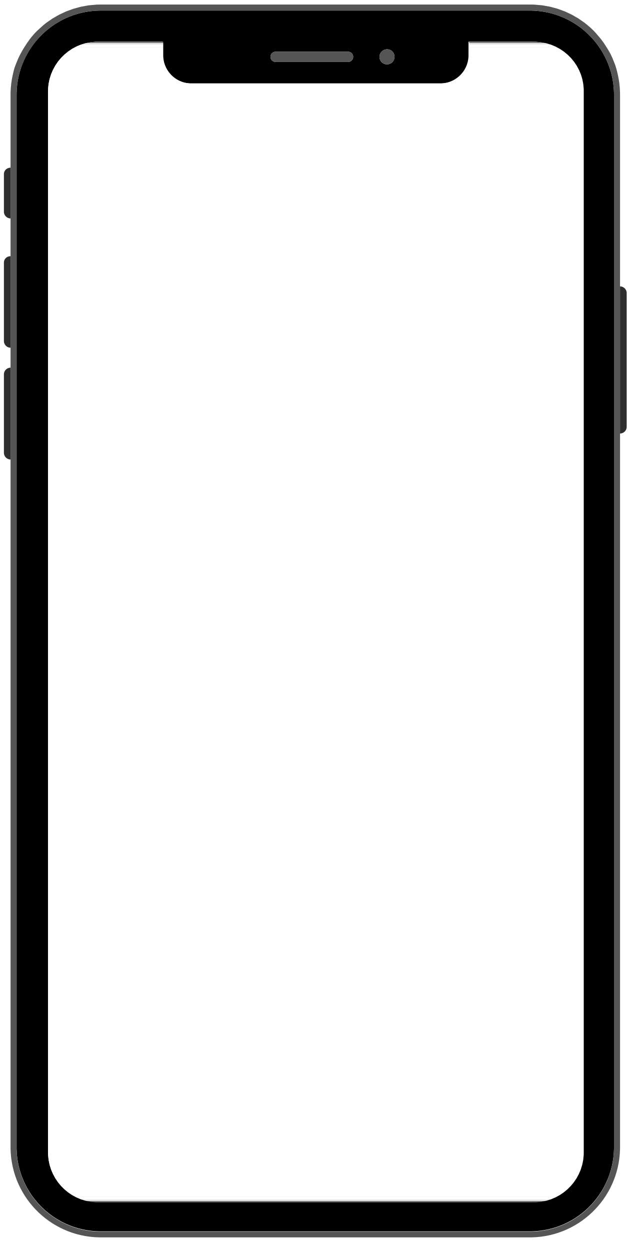


| N° | Issue description. | Checking Status. |
|---|---|---|
| 1 | Compiler Errors | Passed |
| 2 | Race conditions and Reentrancy. Cross-function race conditions. | Passed |
| 3 | Possible delays in data delivery. | Passed |
| 4 | Oracle calls. | Passed |
| 5 | Front running. | Passed |
| 6 | Timestamp dependence. | Passed |
| 7 | Integer Overflow and Underflow. | Passed |
| 8 | DoS with Revert. | Passed |
| 9 | DoS with block gas limit. | Passed |
| 10 | Methods execution permissions. | Passed |
| 11 | Economy model. | Passed |
| 12 | The impact of the exchange rate on the logic. | Passed |
| 13 | Private user data leaks. | Passed |
| 14 | Malicious Event log | Passed |
| 15 | Scoping and Declarations. | Passed |
| 16 | Uninitialized storage pointers. | Passed |
| 17 | Arithmetic accuracy. | Passed |
| 18 | Design Logic. | Passed |
| 19 | Cross-function race conditions. | Passed |
| 20 | Safe Zeppelin module | Passed |
| 21 | Fallback function securitv. | Passed |

# Audit Result:

## PASSED

# WEBSITE - OVERVIEW



## 🕒 Domain Registration:

N/A

## SSL CERTIFICATE: N/A

(https://www.ssllabs.com)

# SOCIAL MEDIA - OVERVIEW

| | |
|---|---|
| ✈ | N/A |
| 🐦 | N/A |
| 🌐 | N/A |